**UE20CS301 DATABASE MANAGEMENT SYSTEM (DBMS)**

**MINI PROJECT REPORT**

**ELECTION MANAGEMENT SYSTEM**

**SUBMITTED BY :**

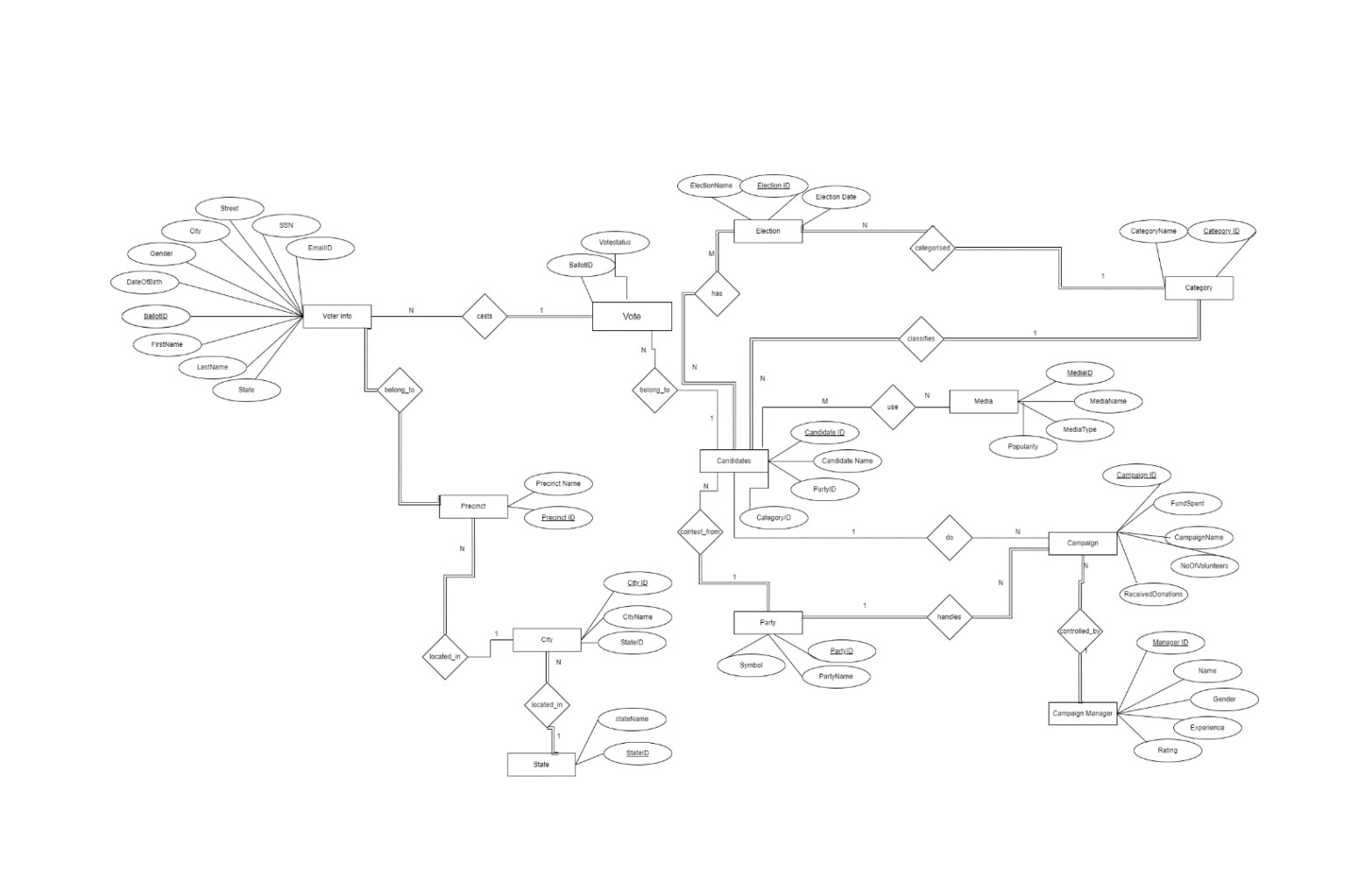
**REHAN AHMED(PES1UG22CS833)**

**SABUJ DAS(PES1UG22CS834)**

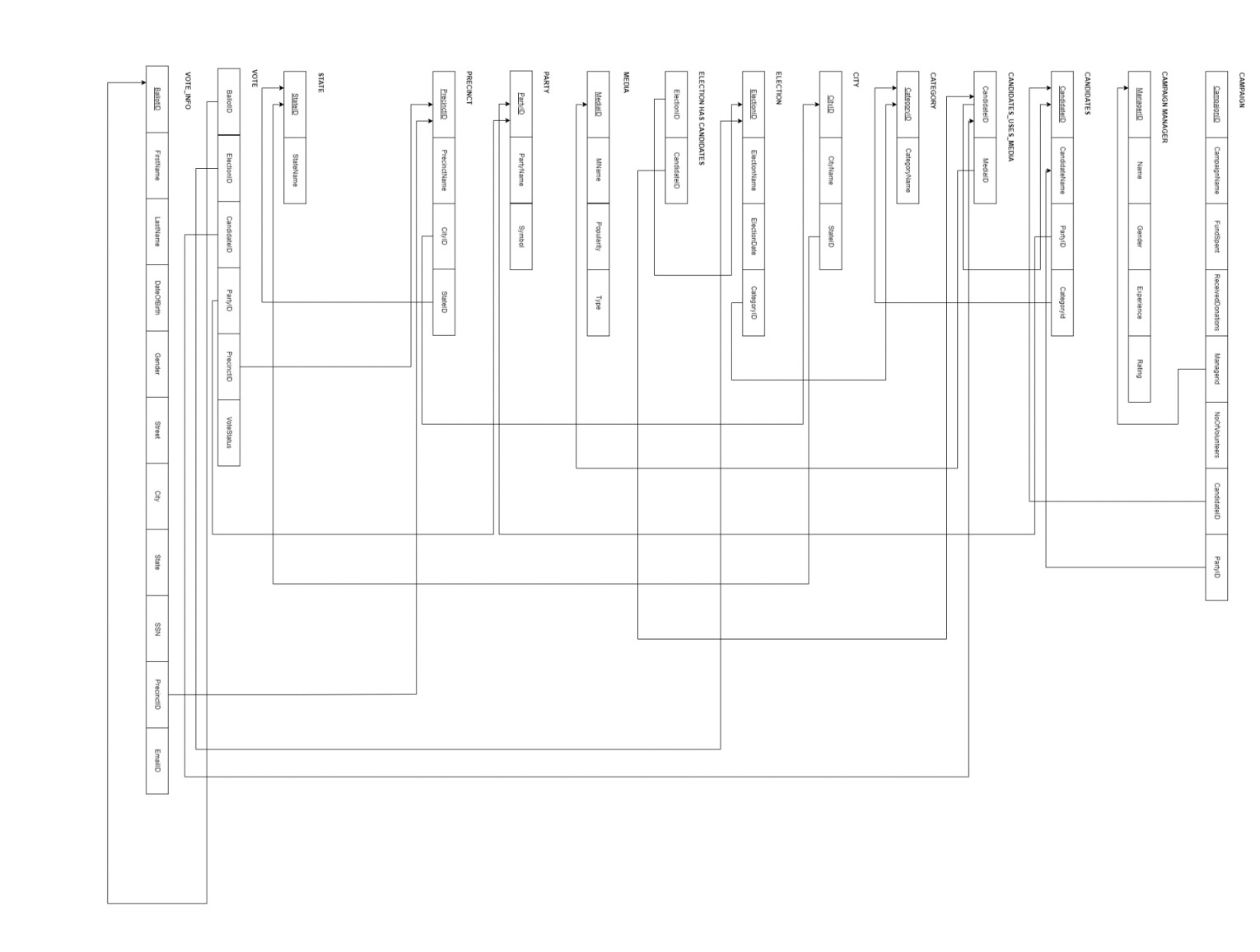
**V SEMESTER**

**SECTION: H**

**ER DIAGRAM**

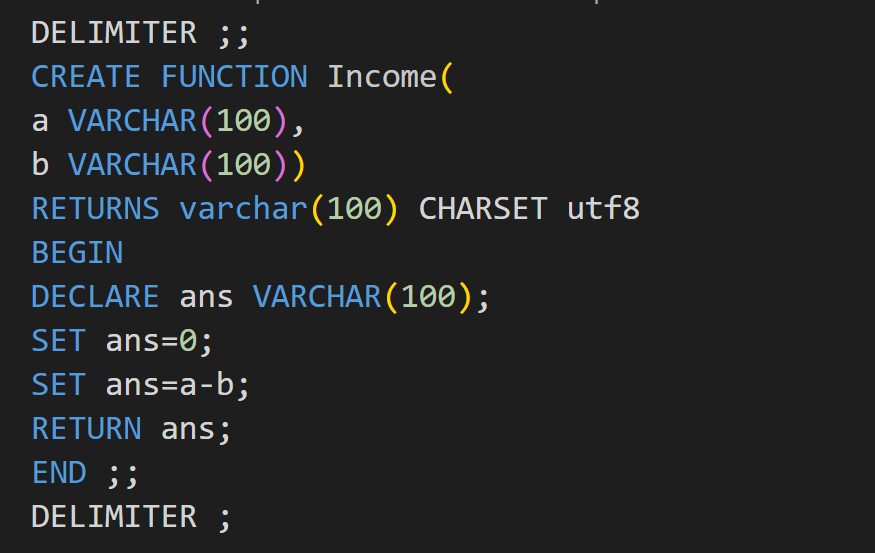


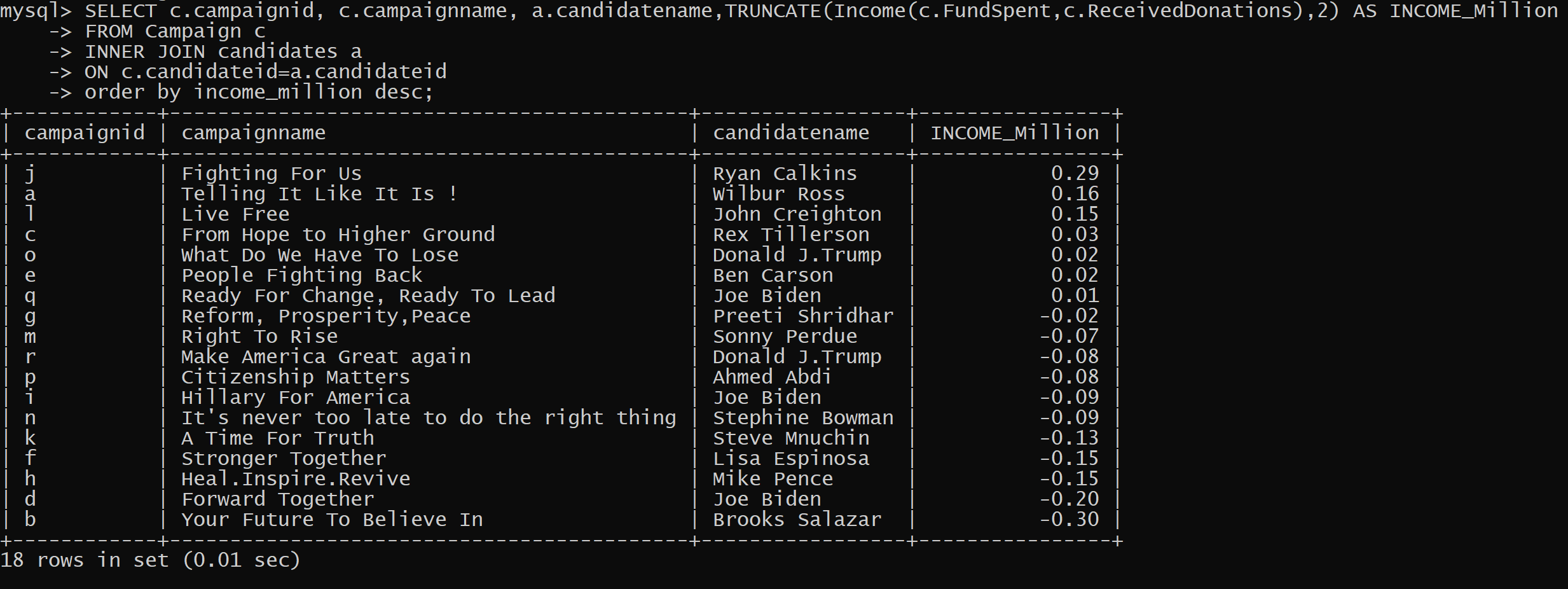
**RELATIONAL SCHEMA**



**FUNCTIONS**

FUNCTION TO DETERMINE THE INCOME GENERATED BY A CAMPAIGN

****

****

**JOINS**

Query to retrieve only the ballot numbers of the voters who have voted

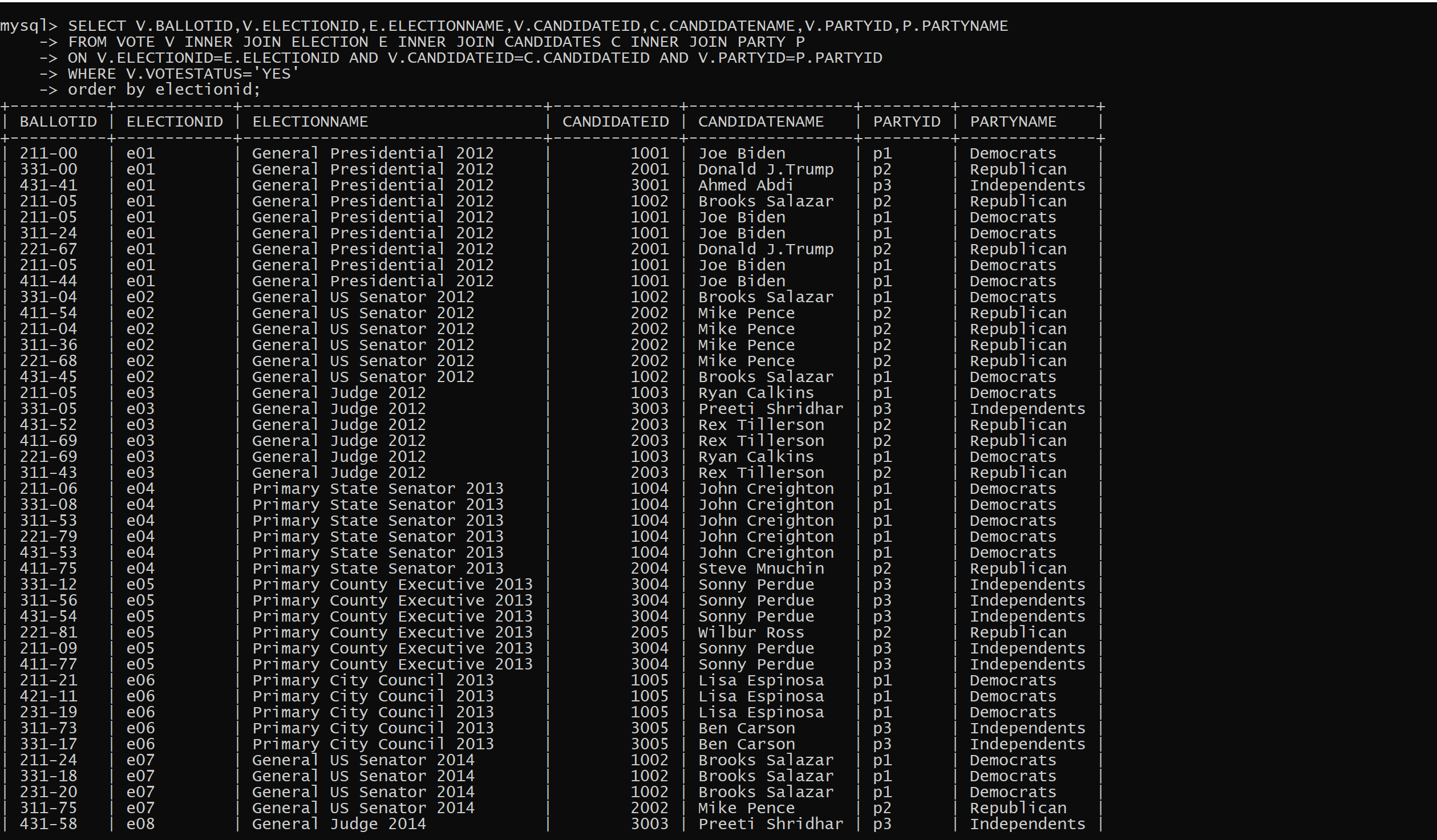
SELECT V.BALLOTID,V.ELECTIONID,E.ELECTIONNAME,V.CANDIDATEID,C.CANDIDATENAME,V.PARTYID,P.PARTYNAME

FROM VOTE V INNER JOIN ELECTION E INNER JOIN CANDIDATES C INNER JOIN PARTY P

ON V.ELECTIONID=E.ELECTIONID AND V.CANDIDATEID=C.CANDIDATEID AND V.PARTYID=P.PARTYID

WHERE V.VOTESTATUS='YES'

order by electionid;

****

Query to get total no. of votes received by each candidate in each election

SELECT v.electionid,

       e.electionname,

       v.candidateid,

       c.candidatename,

       p.PartyName,

       COUNT(v.candidateid) AS Total\_Votes

FROM vote v

INNER JOIN Candidates c

   ON v.CandidateID = c.candidateid

INNER JOIN Election e

   ON v.ElectionID = e.electionid

INNER JOIN Party p

   ON v.partyid = p.partyid

WHERE v.candidateid IS NOT NULL

  AND v.VoteStatus = 'yes'

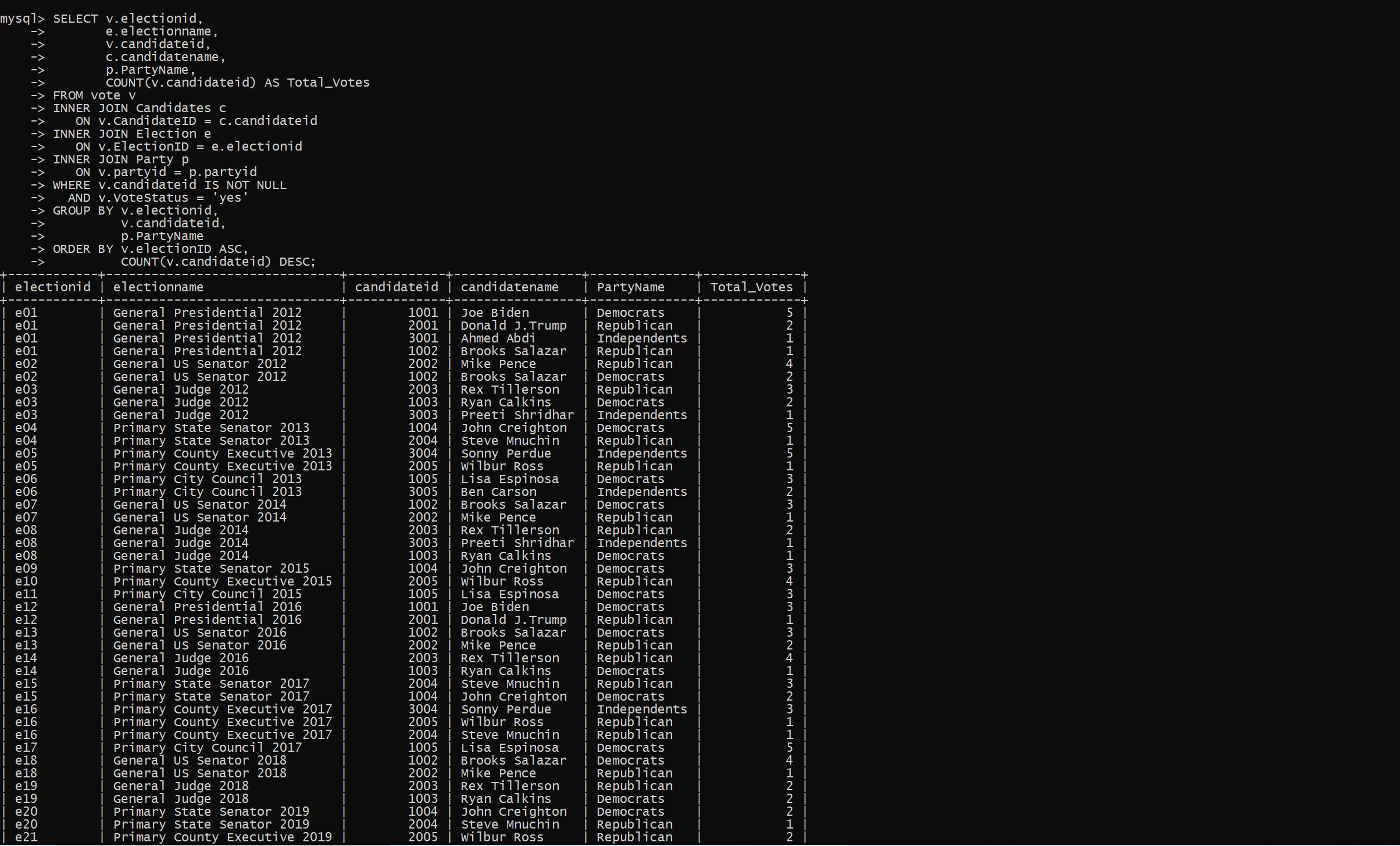
GROUP BY v.electionid,

         v.candidateid,

         p.PartyName

ORDER BY v.electionID ASC,

         COUNT(v.candidateid) DESC;



**VIEWS**

Votes for every candidate

CREATE VIEW Winner AS

SELECT v.electionid,e.electionname, v.candidateid, c.candidatename, p.partyname, count(v.candidateid) AS Total\_Votes

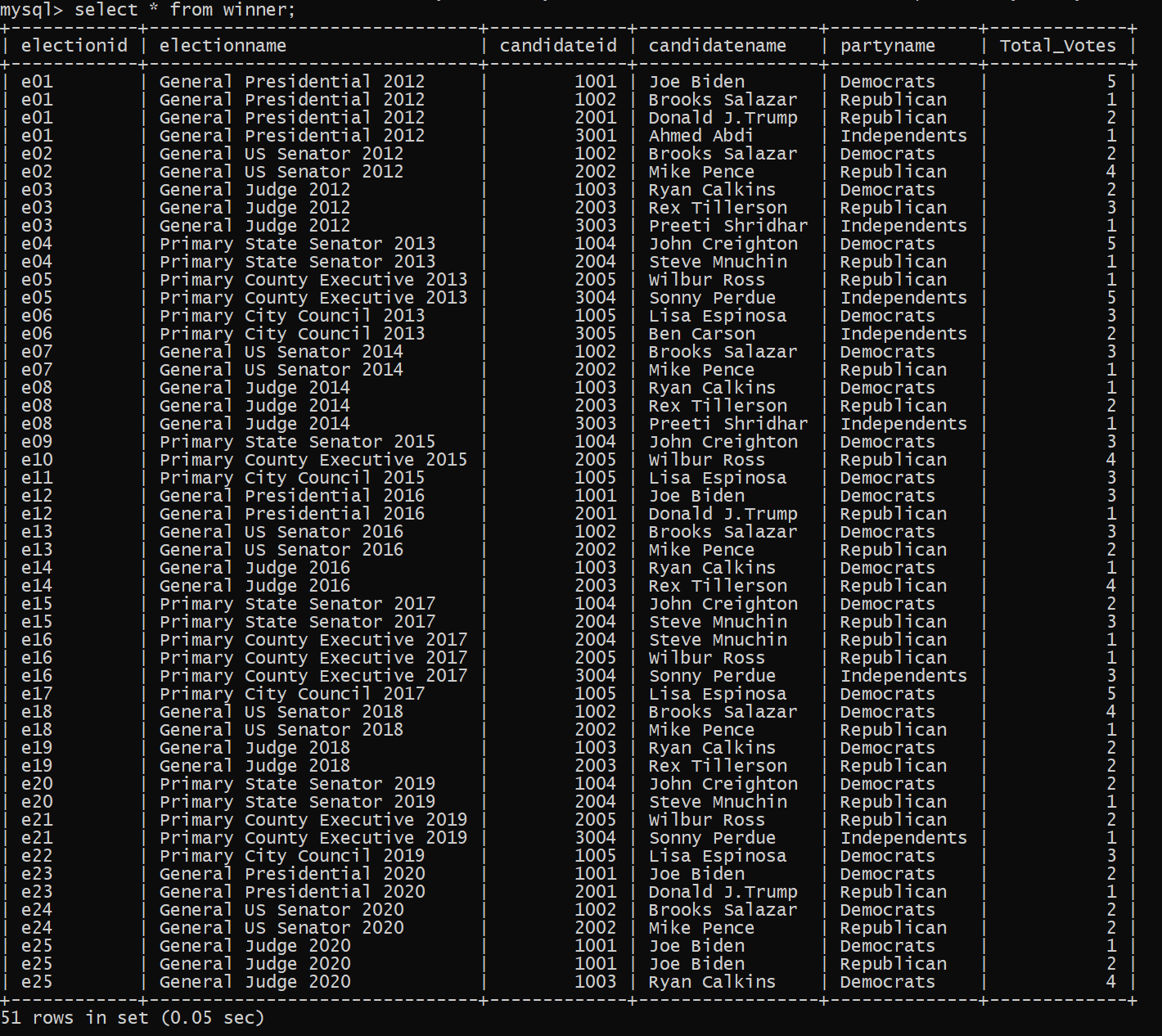
FROM vote v INNER JOIN Candidates c INNER JOIN Election e INNER JOIN Party p

ON v.ElectionID=e.electionid AND v.CandidateID=c.candidateid AND v.partyid=p.partyid

WHERE v.candidateid is NOT NULL AND v.VoteStatus='yes'

GROUP BY v.electionid, v.CandidateID

ORDER BY v.electionID;



Total voters for every election

CREATE VIEW totalvoters

   AS

   select v.electionid, e.electionname, count(v.ballotid) as Total\_votes

   from vote v INNER JOIN election e

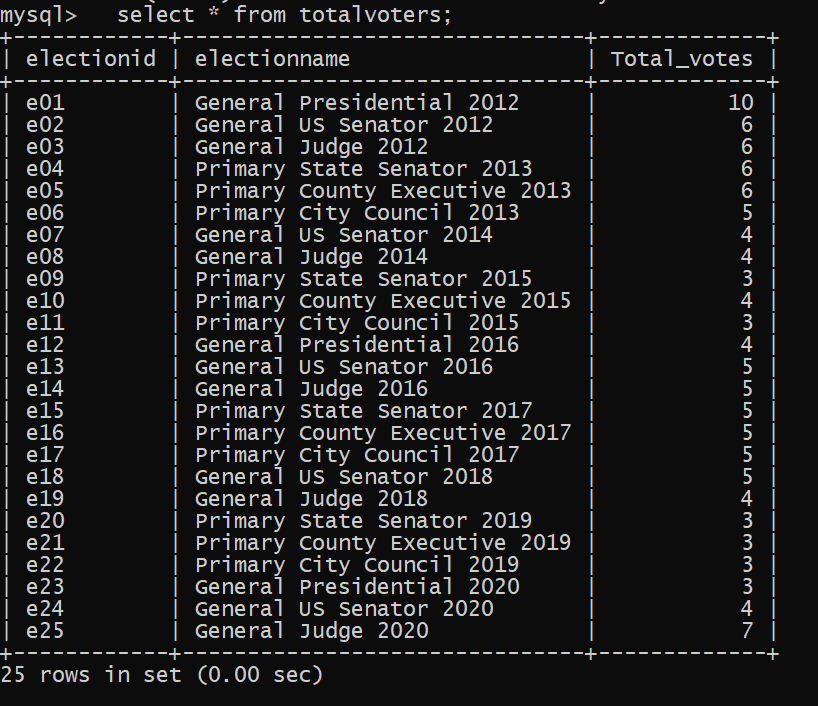
   ON v.electionid=e.electionid

   where v.votestatus='yes'

   group by v.electionid

   order by v.electionid;

  select \* from totalvoters;



**PROCEDURE**

Person who won a particular election for a specific candidate

DELIMITER $$

CREATE PROCEDURE campaigndetails(IN v varchar(60))

BEGIN

SELECT c.campaignid, c.campaignname,cm.mediaid,mm.name,c.fundspent,c.receiveddonations,

TRUNCATE(Income(c.FundSpent,c.ReceivedDonations),2)

AS INCOME\_Million,c.managerid,m.name as Manager\_Name,c.noofvolunteers,c.CandidateID,a.candidatename

FROM Campaign c

INNER JOIN candidates\_uses\_media cm

INNER JOIN candidates a

INNER JOIN campaignManager m

INNER JOIN media mm

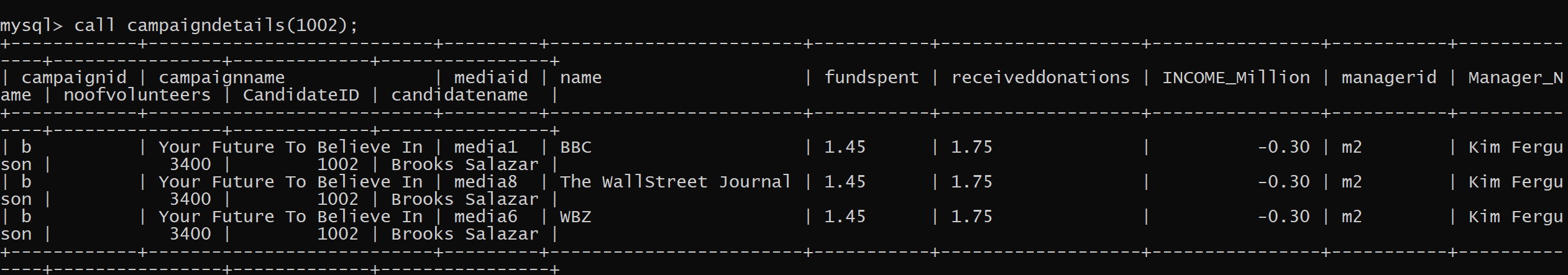
ON c.candidateid=a.candidateid AND m.managerid=c.managerid AND mm.MediaID=cm.MediaID AND cm.candidateid=a.CandidateID

where c.candidateid=v

order by campaignid;

END $$

DELIMITER ;



revenue for each campaign

DELIMITER $$

CREATE PROCEDURE campaignrevenue(IN b varchar(60))

BEGIN

SELECT c.campaignid, c.campaignname, a.candidatename,p.partyname,TRUNCATE(Income(c.FundSpent,c.ReceivedDonations),2) AS INCOME\_Million

FROM Campaign c

INNER JOIN candidates a

INNER JOIN Party p

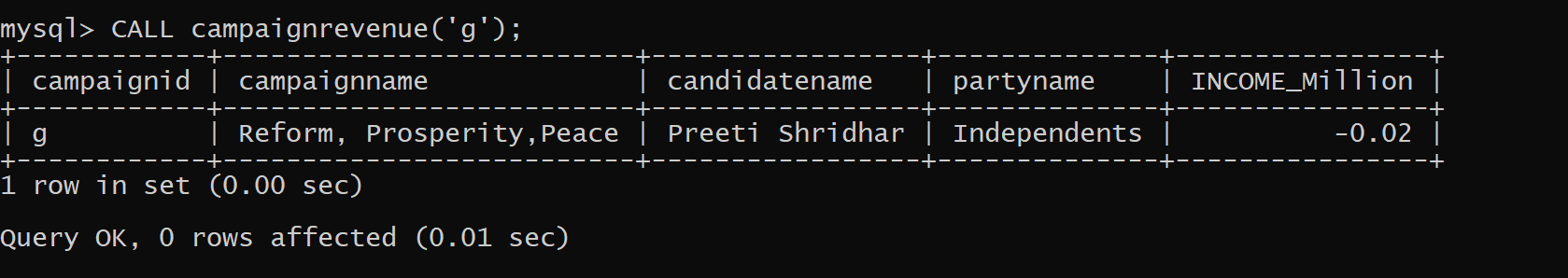
ON c.candidateid=a.candidateid AND p.partyid=a.PartyID

WHERE Campaignid= b

order by income\_million desc;

END $$

DELIMITER ;



**TRIGGGER**

DELIMITER $$

CREATE TRIGGER triggervoter

BEFORE

INSERT ON Voter\_Info

FOR EACH ROW BEGIN

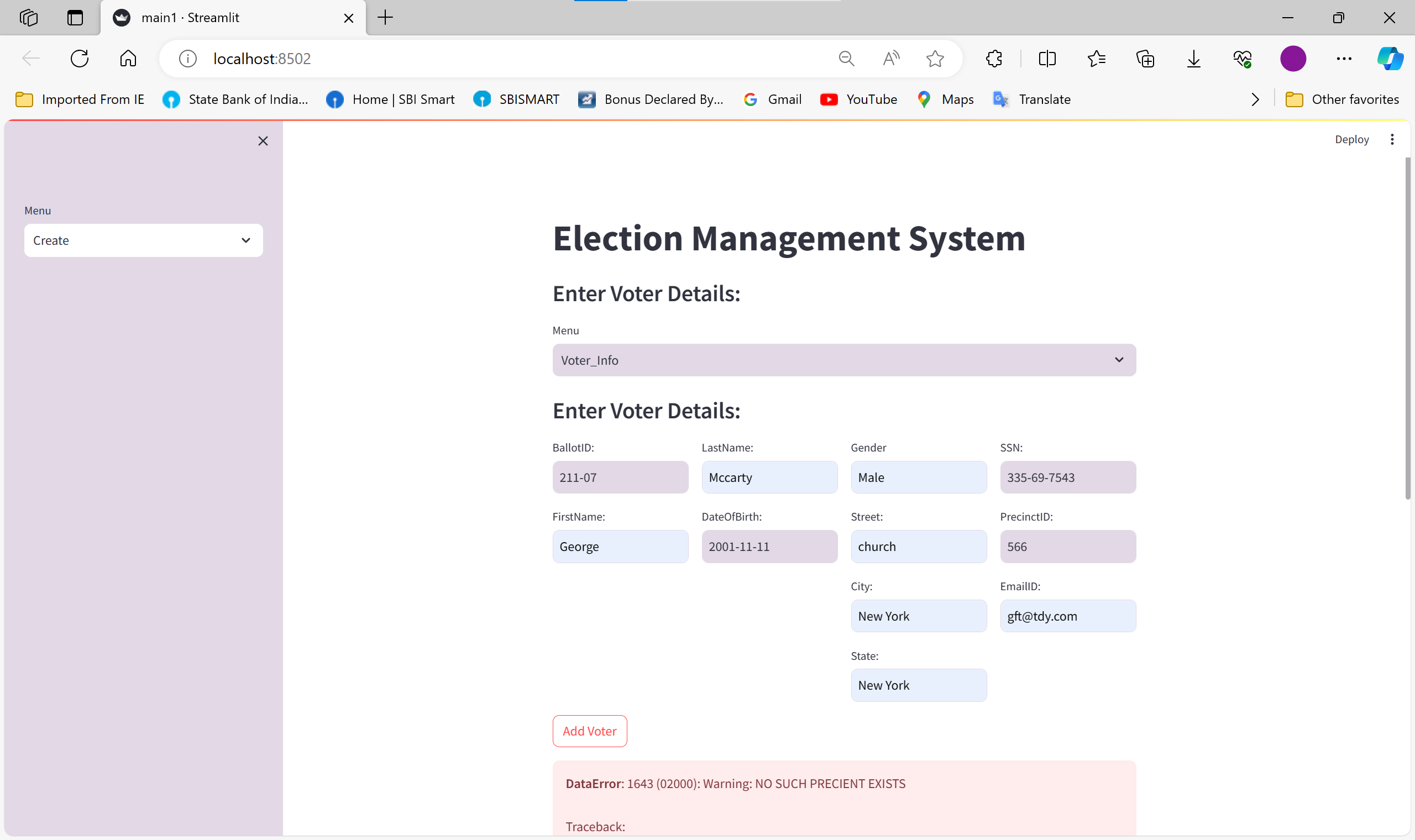
    IF (NEW.precinctid NOT IN (select precinctid from precinct)) THEN

        SIGNAL SQLSTATE '02000' SET MESSAGE\_TEXT = 'Warning: NO SUCH PRECIENT EXISTS';

    END IF;

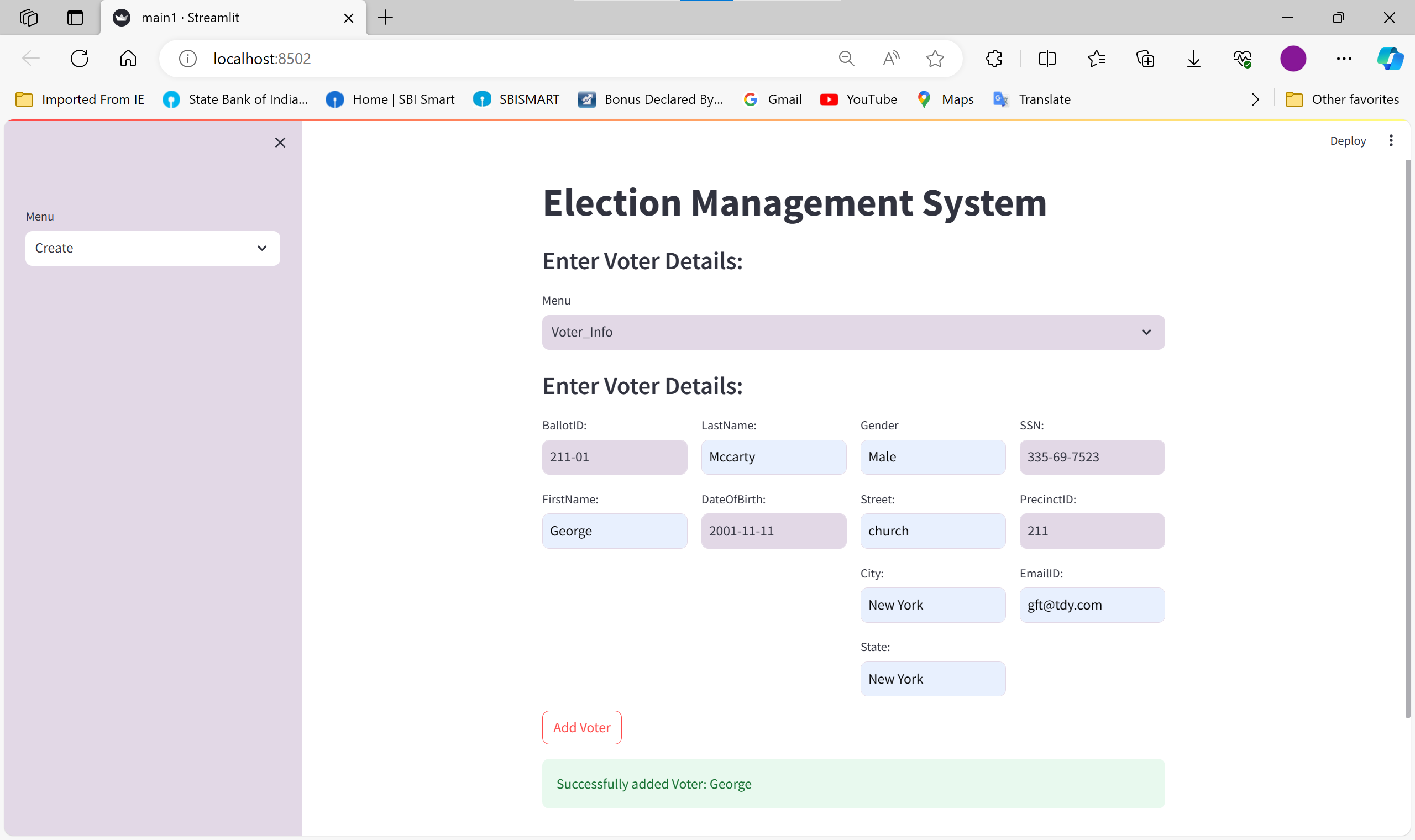
END$$

DELIMITER ;

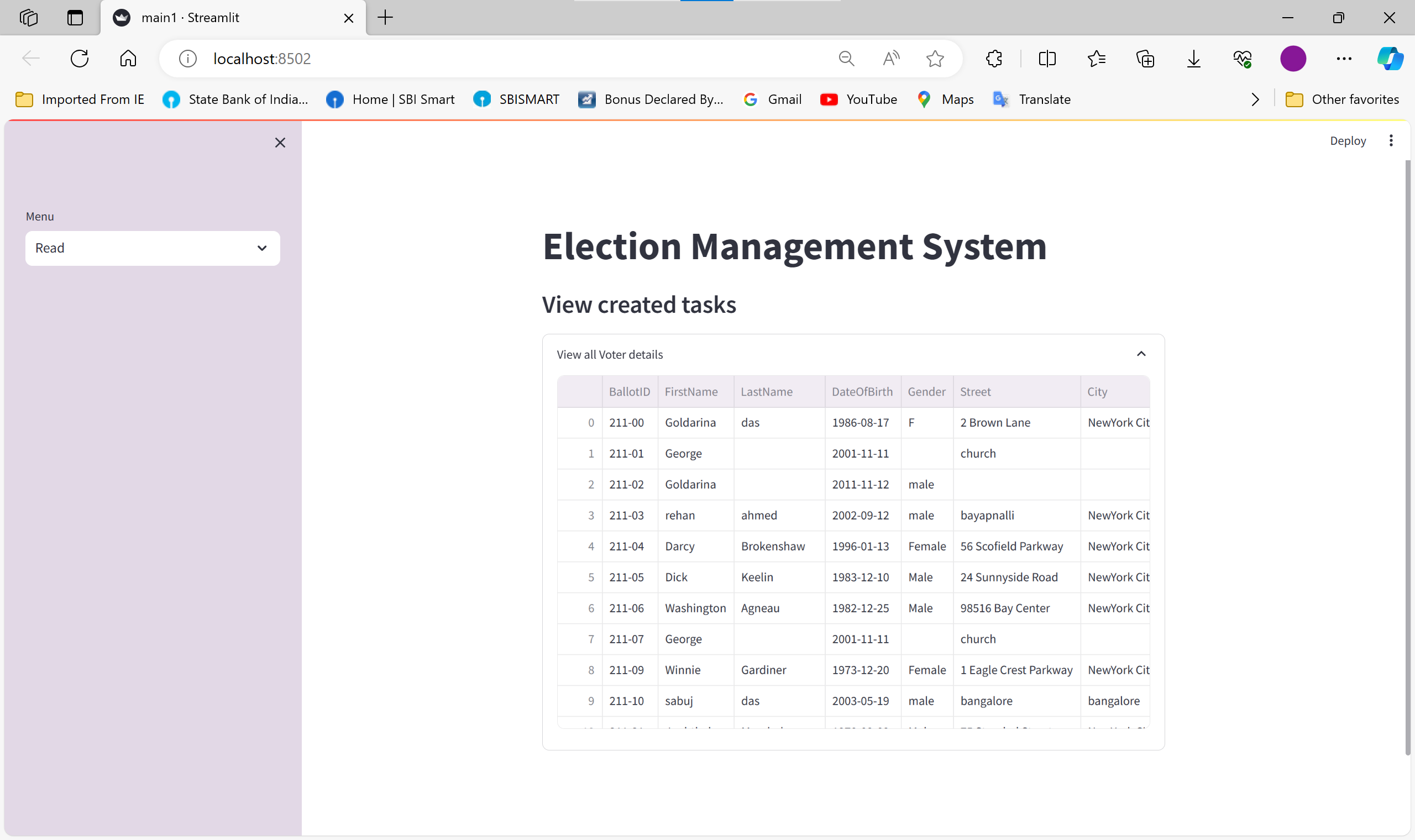
****

**CRUD USING STREAMLT**

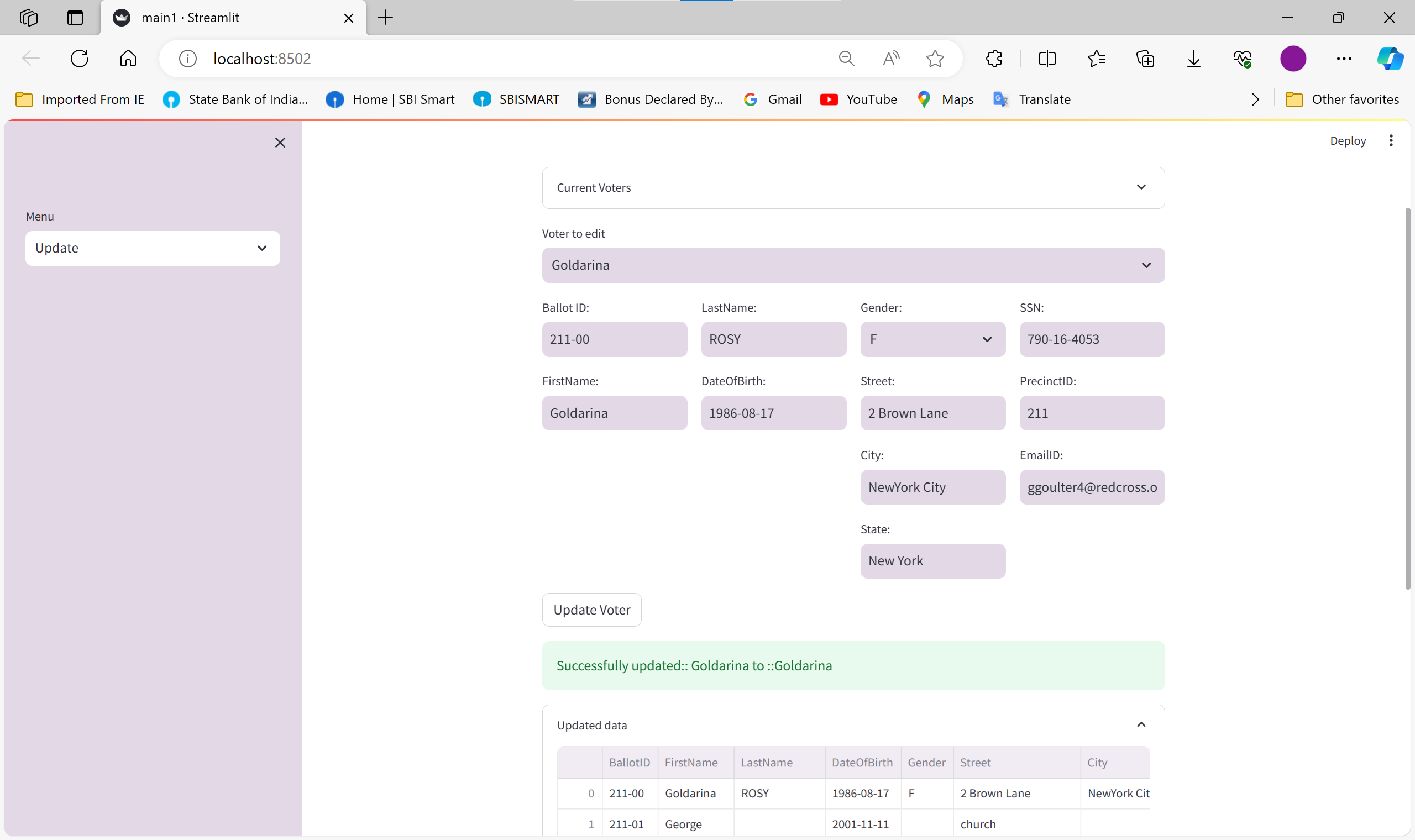
**CREATE**

****

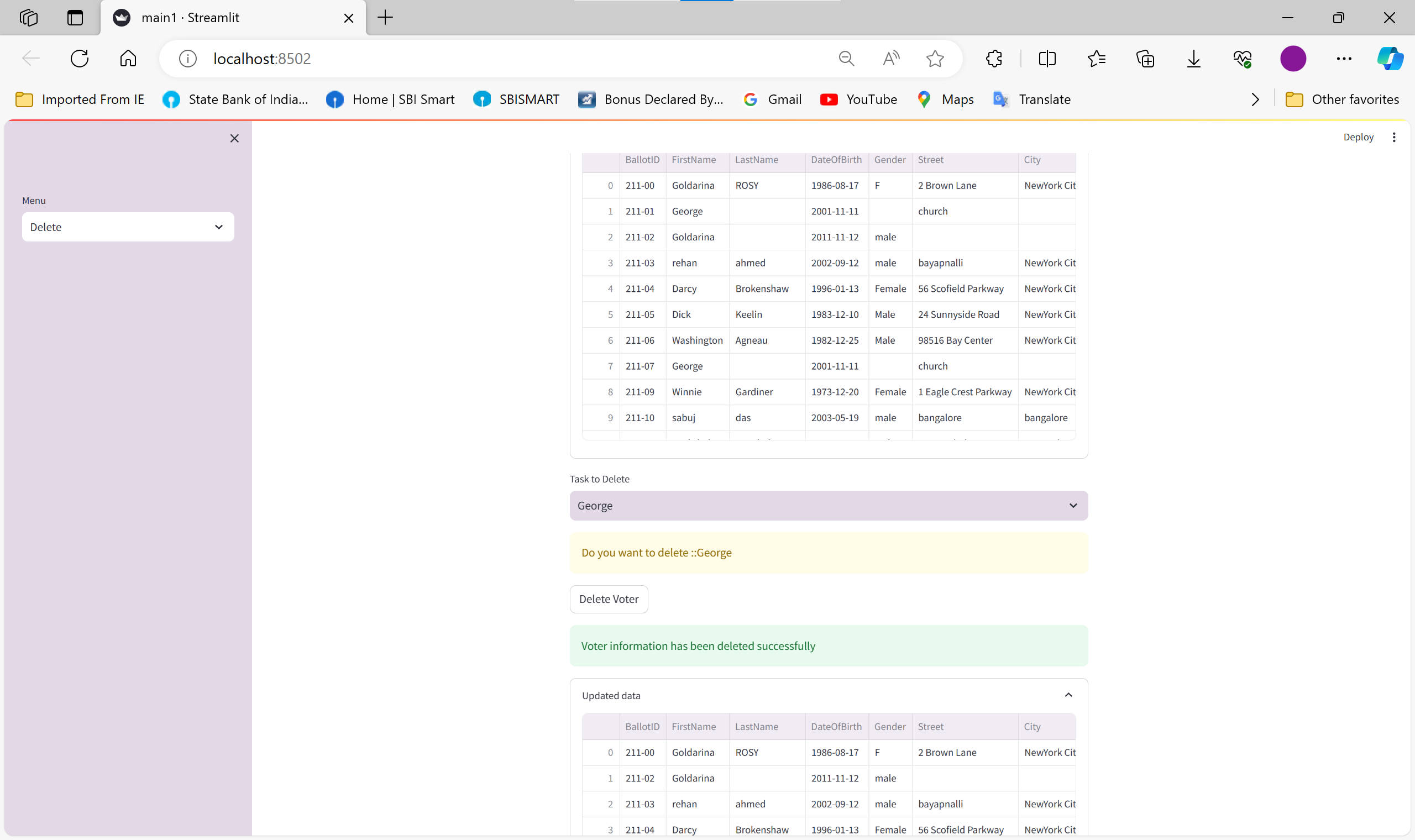
**READ**

****

**UPDATE**

****

**DELETE**

****